FACSIMILE COVER PAGE
(THIS COVER PAGE + 22 PAGES)

Today's Date: October 1, 2006

To: **Examiner K. Tang**, Art Unit: 2195 FAX: (571) 273-8300

From: Frederick E. Cooperrider #36,769
McGinn IP Law Group, PLLC
Ph: (703) 761-2377

In re Application of Ditlow, et al

Serial No.: **09/943,829**

For: METHOD AND APPARATUS TO MANAGE MULTI-COMPUTER SUPPLY

Contents: 1. (Revised) Second Supplemental Appeal Brief (22 pages), in response to the Notification of Non-Compliant Appeal Brief mailed on September 1, 2006

CERTIFICATION OF TRANSMISSION

I certify that I transmitted via facsimile to (571) 273-8300 this Revised Second Supplemental Appeal Brief to Examiner K. Tang on October 1, 2006.

Frederick E. Cooperrider
Reg. No. 36,769

RECEIVED
CENTRAL FAX CENTER
OCT 0 1 2006

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of

Ditlow, et al.

Serial No.: 09/943,829 Group Art Unit: 2195

Filed: August 31, 2001 Examiner: Tang, K.

For: **METHOD AND APPARATUS TO MANAGE MULTI-COMPUTER SUPPLY**

Commissioner of Patents
Alexanderia, VA 22313-1450

APPELLANTS' SECOND SUPPLEMENTAL BRIEF ON APPEAL

Sir:

In response to the Office Action mailed on May 18, 2006, Appellants continue to respectfully appeal the rejection of claims 1-20 in the Office Action dated April 19, 2005. A Notice of Appeal was timely filed on July 19, 2005, and a Brief on Appeal was timely filed on September 19, 2005.

In an Office Action mailed on November 29, 2005, the Examiner re-opened prosecution, citing new references, US Patent Application Publication US 2002/0059625 to Kurauchi, previously-cited US Patent 6,105,053 to Kimmel et al., and newly-cited US Patent 6,016,503 to Overby et al. Appellants evaluated this new rejection and, believing that it fails to further prosecution in any meaningful manner, filed a Petition Under 37 CFR §1.181 to Re-instate Appeal on February 28, 2006, along with a Supplemental Brief to address the new rejection based on Kurauchi.

In the Office Action mailed on May 18, 2006, the Examiner again re-opened prosecution, citing a new reference, US Patent 6,601,084 to Bhaskaran et al. Appellants have evaluated this new rejection and believe it fails to meet the initial burden of a *prima*

facie rejection. Therefore, Appellants do not believe that this new rejection furthers prosecution and have filed herewith a second Petition Under 37 CFR §1.181 to Re-instate Appeal.

This Second Supplemental Brief is submitted in response to this latest Office Action, explaining how the new rejection in this Office Action fails to meet the initial burden of a prima facie rejection and is, therefore, no better than the previous two rejections currently in the process of Appeal.

I. REAL PARTY IN INTEREST

The real party in interest is International Business Machines Corporation, assignee of 100% interest of the above-referenced patent application.

II. RELATED APPEALS AND INTERFERENCES

There are no other appeals or interferences known to Appellants, Appellants' legal representative or Assignee, which would directly affect or be directly affected by or have a bearing on the Board's decision in this appeal.

III. STATUS OF CLAIMS

Claims 1-20, all of the claims presently pending in the application, stand rejected on prior art grounds.

More specifically, claims 1-4, 6-10, 12-16, and 18-20 stand rejected under 35 USC §103(a) as allegedly unpatentable over US Patent 6,601,084 to Bhaskaran et al, further in view of US Patent Application US 2002/0059625 to Kurauchi, and claims 5, 11, and 17 stand rejected under 35 USC §103(a) as unpatentable over Bhaskaran/Kurauchi, further in view of US Patent 6,016,503 to Oberby et al.

IV. STATUS OF AMENDMENTS

An Amendment Under 37 CFR §1.116 was filed on June 20, 2005, although no claim amendments were included therein. Therefore, the version of the claims in the Appendix reflects the claim amendments of the Amendment Under 37 CFR §1.111 filed on January 4, 2005.

In the Advisory Action dated July 21, 2005, the Examiner indicated that the arguments in the Amendment Under 37 CFR §1.116 were not persuasive and that the rejections based on Robertazzi were maintained as based on the Examiner's "... broadest reasonable interpretation consistent with the specification."

A Notice of Appeal was filed on August 19, 2005, and an Appeal Brief was filed on September 19, 2005.

In an Office Action mailed on November 29, 2005, the Examiner re-opened prosecution, citing new references, US Patent Application Publication US 2002/0059625 to Kurauchi, previously-cited US Patent 6,105,053 to Kimmel et al., and newly-cited US Patent 6,016,503 to Overby et al.

A Petition Under 37 CFR §1.181 to Re-instate Appeal on February 28, 2006, along with a Supplemental Appeal Brief to address the new rejection based on Kurauchi.

In the Office Action mailed on May 18, 2006, the Examiner again re-opened prosecution, citing a new reference, US Patent 6,601,084 to Bhaskaran et al. Appellants have again evaluated this new rejection and do not believe that it furthers prosecution and have filed herewith a second Petition Under 37 CFR §1.181 to Re-instate Appeal, along with this second Supplemental Appeal Brief.

A Notification of Noncompliant Appeal Brief was mailed on September 1, 2006. This revised version of the Appeal Brief is submitted in compliance thereto.

V. SUMMARY OF CLAIMED SUBJECT MATTER

Appellants' invention, as disclosed and claimed in independent claim 1, is directed to a computer-implemented method for determining a listing of hosts on a network to

perform a parallel application (lines 15-16 of page 1, lines 16-19 of page 4, lines 1-3 of page 6, Figure 1), including determining a listing of all possible hosts on the network for performing the parallel application (lines 16-18 of page 1, line 19 of page 6 through line 1 of page 7). For each of the possible hosts (line 2 of page 18) a current capacity and a current utilization is determined and a difference between the current capacity and the current utilization is calculated (lines 2-4 of page 7, line12 of page 13, line 3 of page 18 through line 9 of page 19, equation [4] in Figure 7). A listing of hosts is selected from the listing of all possible hosts, based on sorting the calculated differences (lines 18-19 of page 1, lines 4-5 of page 7, lines 6-21 of page 24, Equation [6a] of Figure 7).

The remaining independent claims have similar language.

Independent Claim 1

1. (Rejected) A computer-implemented method determining a listing of hosts on a network to perform a parallel application (lines 15-16 of page 1, lines 12-14 of page 2, lines 16-19 of page 4, lines 1-3 of page 6, Figure 1), said method comprising:

determining a listing of all possible hosts on said network for performing said parallel application (lines 16-18 of page 1, line 19 of page 6 through line 1 of page 7, line 18 of page 17 through line 2 of page 18, see description of "capacity subroutine" 61 at line 14 of page 21 through line 17 of page 22);

determining, for each of said possible hosts, a current capacity and a current utilization (line 4 of page 18 through line 6 of page 19, see description of "capacity subroutine" 61 at line 14 of page 21 through line 17 of page 22 and of "supply subroutine" 62 at line 18 of page 22 through line 3 of page 23);

calculating, for each of said possible hosts, a difference between said current capacity and said current utilization (lines 2-4 of page 7, line12 of page 13, line 3 of page 18 through line 9 of page 19, equation (14) on page 19, equation [4] in Figure 7, see description of "supply subroutine" 62 at line 18 of page 22 through line 3 of page 23); and

selecting from said listing of all possible hosts a listing of hosts based on sorting said calculated differences (lines 18-19 of page 1, lines 4-5 of page 7, lines 6-21 of page 24, Equation [6a] of Figure 7, see description of the "Ownership module" 92 at lines 6-9 of page 24).

Independent Claim 7

7. (Rejected) A signal-bearing medium tangibly embodying a program of machine-readable instructions executable by a digital processing apparatus to determine a listing of hosts on a network to perform a parallel application (lines 15-16 of page 1, lines 12-14 of page 2, lines 16-19 of page 4, lines 1-3 of page 6, Figure 1), said machine-readable instructions comprising:

determining a listing of all possible hosts on said network for performing said parallel application (lines 16-18 of page 1, line 19 of page 6 through line 1 of page 7, line 18 of page 17 through line 2 of page 18, see description of "capacity subroutine" 61 at line 14 of page 21 through line 17 of page 22);

determining, for each of said possible hosts, a current capacity and a current utilization (line 4 of page 18 through line 6 of page 19, see description of "capacity subroutine" 61 at line 14 of page 21 through line 17 of page 22 and of "supply subroutine" 62 at line 18 of page 22 through line 3 of page 23);

calculating, for each of said possible hosts, a difference between said current capacity and said current utilization (lines 2-4 of page 7, line12 of page 13, line 3 of page 18 through line 9 of page 19, equation (14) on page 19, equation [4] in Figure 7, see description of "supply subroutine" 62 at line 18 of page 22 through line 3 of page 23); and

selecting from said listing of all possible hosts a listing of hosts based on sorting said calculated differences (lines 18-19 of page 1, lines 4-5 of page 7, lines 6-21 of page 24, Equation [6a] of Figure 7, see description of "supply subroutine" 62 at line 18 of page 22 through line 3 of page 23).

Independent Claim 13

13. (Rejected) A computer network having a plurality of computation resources and an operating system for executing a target parallel application process using at least a subset of said plurality of computation resources, wherein said network includes a method to determine a listing of said computation resources to perform said target parallel application process (lines 15-16 of page 1, lines 12-14 of page 2, lines 16-19 of page 4, lines 1-3 of page 6, Figure 1) , said method comprising:

determining a listing of all possible said computation resources on said network for performing said parallel application (lines 16-18 of page 1, line 19 of page 6 through line 1 of page 7, line 18 of page 17 through line 2 of page 18, see description of "capacity subroutine" 61 at line 14 of page 21 through line 17 of page 22);

determining, for each of said possible computation resources, a current capacity and a current utilization (line 4 of page 18 through line 6 of page 19, see description of "capacity subroutine" 61 at line 14 of page 21 through line 17 of page 22 and of "supply subroutine" 62 at line 18 of page 22 through line 3 of page 23);

calculating, for each of said possible computation resources, a difference between said current capacity and said current utilization (lines 2-4 of page 7, line12 of page 13, line 3 of page 18 through line 9 of page 19, equation (14) on page 19, equation [4] in Figure 7, see description of "supply subroutine" 62 at line 18 of page 22 through line 3 of page 23); and

selecting from said listing of all possible computation resources a listing of computation resources based on sorting said calculated differences as said at least a subset of said plurality of computation resources to execute said target parallel application process (lines 18-19 of page 1, lines 4-5 of page 7, lines 6-21 of page

24, Equation [6a] of Figure 7, see description of "supply subroutine" 62 at line 18 of page 22 through line 3 of page 23).

Independent Claim 19

19. (Rejected) A computer network having a plurality of computation resources and an operating system for executing a target parallel application process using at least a subset of said plurality of computation resources, wherein said network includes a method to determine a listing of said computation resources to perform said target parallel application process (lines 15-16 of page 1, lines 12-14 of page 2, lines 16-19 of page 4, lines 1-3 of page 6, Figure 1), said method comprising:

means for determining a listing of all possible said computation resources on said network for performing said parallel application (equation (14) on page 19, also, lines 16-18 of page 1, line 19 of page 6 through line 1 of page 7, line 18 of page 17 through line 2 of page 18);

means for determining, for each of said possible computation resources, a current capacity and a current utilization (see description of "capacity subroutine" 61 at line 14 of page 21 through line 17 of page 22 and of "supply subroutine" 62 at line 18 of page 22 through line 3 of page 23, also, line 4 of page 18 through line 6 of page 19);

means for calculating, for each of said possible computation resources, a difference between said current capacity and said current utilization (see description of "supply subroutine" 62 at line 18 of page 22 through line 3 of page 23, also, lines 2-4 of page 7, line12 of page 13, line 3 of page 18 through line 9 of page 19, equation (14) on page 19, equation [4] in Figure 7); and

means for selecting from said listing of all possible computation resources a listing of computation resources based on sorting said calculated differences to be said at least a subset of said computation resources for executing said target parallel application process (see description of "supply subroutine" 62 at line 18 of page 22

through line 3 of page 23, also, lines 18-19 of page 1, lines 4-5 of page 7, lines 6-21 of page 24, Equation [6a] of Figure 7).

Dependent Claim 5

5. (Rejected) The method of claim 1, wherein said selecting a listing of hosts from said listing of all possible hosts further comprises a quantification of a history of each said possible host and a consideration of said history in said selecting of a listing. (lines 9-12 on page 24, "box" shown in Figure 2 that provides statistical parameters that define the host's history)

Dependent Claim 11

11. (Rejected) The signal-bearing medium of claim 7, wherein said selecting a listing of hosts from said listing of all possible hosts further comprises a quantification of a history of each said possible host and a consideration of said history in said selecting of a listing. (lines 9-12 on page 24, "box" shown in Figure 2 that provides statistical parameters that define the host's history)

Dependent Claim 17

17. (Rejected) The computer network of claim 13, wherein said selecting a listing of computation resources from said listing of all possible computation resources further comprises a quantification of a history of each said possible computation resource and a consideration of said history in said selecting of a listing. (lines 9-12 on page 24, "box" shown in Figure 2 that provides statistical parameters that define the host's history)

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Appellants present the following grounds for rejection for review by the Board of Patent Appeals and Interferences:

1. Claims 1-20 stand rejected under 35 U.S.C. §103(a), as based upon newly-cited US Patent 6,601,084 to Bhaskaran et al. as the primary reference

2. Claims 1-4, 6-10, 12-16, and 18-20 stand rejected under 35 U.S.C. §103(a), as based upon US Patent Publication US 2002/0059625 to Kurauchi as a secondary reference

3. Claims 5, 11, and 17 stand rejected under 35 U.S.C. §103(a), as based upon US Patent 6,016,503 to Overby, Jr. et al. as a second secondary reference

VII. ARGUMENTS

GROUNDS #1: THE REJECTIONS FOR CLAIMS 1-20 AS BASED ON BHASKARAN AS THE PRIMARY REFERENCE

The Examiner alleges that the present invention defined by claims 1-4, 6-10, 12-16, and 18-20 are rendered obvious when newly-cited primary reference, US Patent 6,601,084 to Bhaskaran is modified by previously-cited US Patent Application US 2002/0059625 to Kurauchi, and claims 5, 11, and 17 are rendered obvious when Bhaskaran/Kurauchi, is further modified by previously-cited US Patent 6,016,503 to Oberby et al.

Appellants' Position of the Rejection Based on Primary Reference Bhaskaran

A. Bhaskaran is non-analogous art, by failing to address a parallel processing application

Appellants first point out that newly-cited primary reference Bhaskaran is directed toward the problem of dynamic load balance for multiple network servers and does not

address the parallel processing environment as required by the independent claims. Therefore, this reference is non-analogous art for the present invention.

As well understood by one having ordinary skill in the art, parallel processing involves, in general, the distribution of processing tasks to be executed independently on separate CPUs. This generalized description is confirmed by the following definition from the 6th Edition of the "Dictionary of Computer and Internet Terms", D. A. Downing, Ph.D., et al., 1998, Barron's Educational Series, Inc.:

"PARALLEL PROCESSING computation carried out at the same time on different CPUs, or on a CPU that can execute more than one instruction at the exact same time. By contrast, most multitasking is accomplished by making a single CPU switch its attention among several tasks. This is called concurrent processing or timesharing."

The description at lines 12-16 of page 2 of the specification confirms this definition: "*Multi-computer processing, hereinafter called multi-processing, involves the use of multiple computers to process a single computational task such as an application program. The term generally refers to the use of a parallel or a distributed computer. The programming of either a parallel or a distributed computer generally is referred to as "parallel programming."*

Appellants submit that the processing described in primary reference Bhaskaran is not parallel processing, as addressed by the present invention and as required by the plain meaning of the claim language. Simply having the servers on a network operating in parallel or a computer directing a process to balance the loads for the network servers does not bestow the property of "parallel processing." Each computer in Bhaskaran that performs its role as a server is operating independently of the other servers and a computer executing the load balancing method is operating on its own.

In contrast, parallel processing would require that more than one of the servers and/or the computer executing the load balancing method are concurrently involved in a common processing, such that each involved computer performs their respective assigned computing tasks for that common processing. This common processing, with each of several computers performing their respective share of the processing computations in

parallel, is not present in Bhaskaran, since neither the server function nor the load balancing requires that the various computers be executing in parallel a shared processing.

B. Bhaskaran teaches against the technique of the present invention

Moreover, Appellants submit that the method in primary reference Bhaskaran, because of its different purpose of balancing loads for servers on a network, also fails to satisfy the plain meaning of the claim language of the independent claims.

More specifically, the goal in Bhaskaran is that of equalizing the load among all of the servers. Therefore, Bhaskaran teaches against the method described by the plain meaning of the claim limitations, since the load balancing method is based upon first calculating the relative amount of power for each computer in the network, as expressed by equation 1 at line 35 in column 6. This relative power is the basis that each machine is performing its fair share in the network as determined by its relative power.

Because of this dependence on the relative power of each machine, there is no simple difference between current capacity and current utilization, as required to satisfy the plain meaning of the claim language: "... calculating, for each of said possible hosts, a difference between said current capacity and said current utilization ...."

C. Bhaskaran does not need to provide a listing of hosts

Finally, it is also noted that the load balancing Bhaskaran does not satisfy the final claim limitation and cannot be modified to do so, since the goal in this reference is to balance the load evenly through out the network, as explained at lines 42-47 of column 3. Thus, in Bhaskaran all servers will always be performing approximately their fair share as determined based on their relative power level within the network, and, as described at lines 32-38 of column 4, a skew detector determines when a server's performance is outside a tolerable range, and buckets are created to provide additional load to that server.

Thus, in the task being done in Bhaskaran, there is no need to select a listing of hosts, let along make selections based on a sorting, as required by the plain meaning of the claim language: "... selecting from said listing of all possible hosts a listing of hosts based on sorting said calculated differences."

D. Bhaskaran cannot serve as the primary reference

Appellants further submit that modifying primary reference Bhaskaran to select a listing of hosts would defeat the purpose therein of maintaining the work balance within the predetermined tolerance. Therefore, Appellants submit that it would be improper to modify Bhaskaran by either secondary reference currently of record or any other reference in the manner described by the final claim limitation, and, again, Appellants submit that the rejection currently of record fails to meet the initial burden of a *prima facie* rejection.

Hence, turning to the clear language of the claims, in view of the above discussion, in primary reference Bhaskaran, there is no teaching or suggestion of: ".... determining a listing of hosts on a network to perform a parallel application, said method comprising: determining a listing of all possible hosts on said network for performing said parallel application ....", as required by independent claim 1. The remaining independent claims also make reference to the parallel processing environment.

For this reason alone, Appellants submit that the rejection currently of record fails to meet the initial burden of a *prima facie* rejection based upon primary reference Bhaskaran and cannot be maintained and that claims 1-20 would be patentable over Bhaskaran for this reason alone.

GROUNDS # 2: THE REJECTIONS FOR CLAIMS 1-4, 6-10, 12-16, AND 18-20 AS BASED ON KURAUCHI AS A SECONDARY REFERENCE

The Examiner alleges that the present invention defined by claims 1-4, 6-10, 12-16, and 18-20 are rendered obvious when Bhaskaran is modified by Kurauchi. The Examiner alleges that primary reference "*Bhaskaran is silent on sorting (by priority) said calculated differences*", and alleges that secondary reference "*Kurauchi teaches sorting by priority the available amount of resources ([0094], [0173], [0192])*" and that one having ordinary skill in the art would have been motivated to modify Bhaskaran in accordance with Kurauchi "*... because it would improve the utilization efficiency of hardware resources, as stated in Kurauchi (see Abstract and page 1, [0013]).*"

Appellants' Position of the Rejection Based on Secondary Reference Kurauchi

A. Secondary reference Kurachi is non-analogous art

As an initial matter and similar to the argument above for primary reference Bhaskaran, the simple broadcast system of secondary reference Kurauchi does not address a parallel processing problem. Rather, in Kurauchi, each hardware module performs its respective and independent task, designed as an independent specific broadcasting task assigned by the CPU. When its task is completed, the hardware modules of Kurauchi have no processing result that is a partial computation of a parallel processing problem to be returned to a central location to be merged into results from the other modules.

Therefore, the task assignment process in this reference is merely that of ensuring that the various independent tasks related to broadcasting are appropriately assigned to an available hardware module within the limits of the nonlinear editing device's resources. This assignment process is not in any way related to assigning tasks for a parallel processing problem, as this term is understood in the art, even if the broadcast tasks are separately being executed in parallel by the various hardware modules.

Therefore, Appellants submit that secondary reference Kurauchi is also non-analogous art to that of the present invention that addresses an assignment of tasks for a parallel processing problem and that, therefore, even if Kurauchi were to be combined with Bhaskaran, the combination would not overcome the deficiencies identified above for primary reference Bhaskaran.

As pointed out in the above discussion for Bhaskaran, the modification of the primary reference to address this deficiency identified by the Examiner upon which secondary reference Kurauchi is relied upon, would defeat the purpose of Bhaskaran and change its principle of operation.

It is further pointed out that Kurauchi addresses the distribution of video data stored in a randomly accessible recording medium, which problem has nothing to do with the problem in primary reference Bhaskaran, that of balancing loads for servers on a network. That is, since Kurauchi is directed to an entirely different problem from Bhaskaran, these two references are also non-analogous to each other.

Therefore, for any of the above reasons, Kurauchi is not properly combinable with Bhaskaran.

B. The rejection has no proper motivation to modify the primary reference

In paragraph 2 beginning on page 2 of the Office Action mailed on May 18, 2006, the Examiner characterizes in the rejection that "*Bhaskaran is silent on sorting (by priority) said calculated differences.*" (line 3 on page 3)

The Examiner continues: "*However, Kurauchi teaches sorting by priority the available amount of resources ([0094],[0173],[0192]). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine Bhaskaran and Kurauchi because it would improve teh utilization efficiency of hardware resources, as stated in Kurauchi (see Abstract and page 1, [0013].*" (lines 3-7 on page 3 of the Office Action)

Contrary to the Examiner's characterization, primary reference Bhaskaran is not "... silent on sorting ...." Rather, as discussed above, the primary reference clearly teaches against a sorting technique, because of its different environment and different goal.

More specifically, the goal in Bhaskaran is that of equalizing the load among all of the servers that each perform their own tasks independent of other servers (i.e., a non-parallel processing application). Therefore, Bhaskaran teaches against the method described by the plain meaning of the claim limitations, since the load balancing method is based upon first calculating the relative amount of power for each computer in the network, as expressed by equation 1 at line 35 in column 6. This relative power is the basis that each machine is performing its fair share in the network as determined by its relative power.

GROUNDS # 3: THE REJECTIONS FOR CLAIMS 5, 11, AND 17 AS BASED ON OVERBY AS A SECOND SECONDARY REFERENCE

Relative to Overby, in the exemplary embodiment of the present invention, the utilization is based on CPU, memory, temporary files, and cache for each host machine and

how these properties vary over time. Utilization is combined with capacity, normalized and used with a priority when selecting hosts that will solve a parallel problem.

In contrast, Overby's utilization applies to a shared resource, for example, the amount of shared memory used, and this utilization is used to manage the shared resource. This is an entirely different concept from that of managing resources on separate machines, let alone doing so for a parallel processing environment.

Therefore, Appellants submit that Overby adds nothing to overcome the basic deficiencies of Bhaskaran and/or Kurauchi.

More important, even if Overby teaches an historical technique, the technique in primary reference Bhaskaran is based entirely upon balancing the load at the present moment. It would be improper to incorporate an aspect of history into the selection process of Bhaskaran, since such historical factor would defeat this purpose of balancing the load at the present moment and/or change its principle of operation. Therefore, Overby cannot be combined with Bhaskaran.

CONCLUSION

In view of the foregoing, Appellants submit that claims 1-20, all the claims presently pending in the application, are clearly enabled and patentably distinct from the prior art of record and in condition for allowance. Thus, the Board is respectfully requested to remove all rejections of claims 1-20.

Please charge any deficiencies and/or credit any overpayments necessary to enter this paper to Assignee's Deposit Account number 09-0456.

Respectfully submitted,

Dated: 10/1/06

Frederick E. Cooperrider
Reg. No. 36,769

McGinn Intellectual Property Law Group, PLLC.
8231 Old Courthouse Road, Suite 200
Vienna, VA 22182-3817
(703) 761-4100
Customer Number: 21254
Docket BUR920000146US1

VIII. CLAIMS APPENDIX

Claims, as reflected upon entry of the Amendment Under 37 CFR §1.116 filed on June 20, 2005:

1. (Rejected) A computer-implemented method determining a listing of hosts on a network to perform a parallel application, said method comprising:

determining a listing of all possible hosts on said network for performing said parallel application;

determining, for each of said possible hosts, a current capacity and a current utilization;

calculating, for each of said possible hosts, a difference between said current capacity and said current utilization; and

selecting from said listing of all possible hosts a listing of hosts based on sorting said calculated differences.

2. (Rejected) The method of claim 1, wherein said determination of a listing of processors is itself a parallel processing application.

3. (Rejected) The method of claim 1, wherein said determination of a listing of processors is executed in real time concurrently with said parallel application.

4. (Rejected) The method of claim 1, further comprising:

providing said selected listing of hosts to an operating system controlling an execution of said parallel application.

5. (Rejected) The method of claim 1, wherein said selecting a listing of hosts from said listing of all possible hosts further comprises a quantification of a history of each said possible host and a consideration of said history in said selecting of a listing.

6. (Rejected) The method of claim 1, wherein said calculating a difference between current capacity and a current utilization further comprises:

normalizing said difference.

7. (Rejected) A signal-bearing medium tangibly embodying a program of machine-readable instructions executable by a digital processing apparatus to determine a listing of hosts on a network to perform a parallel application, said machine-readable instructions comprising:

determining a listing of all possible hosts on said network for performing said parallel application;

determining, for each of said possible hosts, a current capacity and a current utilization;

calculating, for each of said possible hosts, a difference between said current capacity and said current utilization; and

selecting from said listing of all possible hosts a listing of hosts based on sorting said calculated differences.

8. (Rejected) The signal-bearing medium of claim 7, wherein said machine-readable instructions are provided to an operating system on said network such that said determination of a listing of processors is itself a parallel processing application.

9. (Rejected) The signal-bearing medium of claim 7, wherein said machine-readable instructions are provided to an operating system on said network such that said determination of a listing of processors is executed in real time concurrently with said parallel application.

10. (Rejected) The signal-bearing medium of claim 7, said machine-readable instructions further comprising:

providing said selected listing of hosts to an operating system controlling an execution of said parallel application.

11. (Rejected) The signal-bearing medium of claim 7, wherein said selecting a listing of hosts from said listing of all possible hosts further comprises a quantification of a history of each said possible host and a consideration of said history in said selecting of a listing.

12. (Rejected) The signal-bearing medium of claim 7, wherein said calculating a difference between current capacity and a current utilization further comprises:

normalizing said difference.

13. (Rejected) A computer network having a plurality of computation resources and an operating system for executing a target parallel application process using at least a subset of said plurality of computation resources, wherein said network includes a method to determine a listing of said computation resources to perform said target parallel application process, said method comprising:

determining a listing of all possible said computation resources on said network for performing said parallel application;

determining, for each of said possible computation resources, a current capacity and a current utilization;

calculating, for each of said possible computation resources, a difference between said current capacity and said current utilization; and

selecting from said listing of all possible computation resources a listing of computation resources based on sorting said calculated differences as said at least a subset of said plurality of computation resources to execute said target parallel application process.

14. (Rejected) The computer network of claim 13, wherein said method interfaces to an operating system on said network such that said determination of a listing of computation resources is itself a parallel processing application.

15. (Rejected) The computer network of claim 13, wherein said method interfaces to an operating system on said network such that said determination of a listing of computation resources is executed in real time concurrently with said parallel application.

16. (Rejected) The computer network of claim 13, said method further comprising:

providing said selected listing of computation resources to an operating system controlling an execution of said parallel application.

17. (Rejected) The computer network of claim 13, wherein said selecting a listing of computation resources from said listing of all possible computation resources further comprises a quantification of a history of each said possible computation resource and a consideration of said history in said selecting of a listing.

18. (Rejected) The computer network of claim 13, wherein said calculating a difference between current capacity and a current utilization further comprises:

normalizing said difference.

19. (Rejected) A computer network having a plurality of computation resources and an operating system for executing a target parallel application process using at least a subset of said plurality of computation resources, wherein said network includes a method to determine a listing of said computation resources to perform said target parallel application process, said method comprising:

means for determining a listing of all possible said computation resources on said network for performing said parallel application;

means for determining, for each of said possible computation resources, a current capacity and a current utilization;

means for calculating, for each of said possible computation resources, a difference between said current capacity and said current utilization; and

means for selecting from said listing of all possible computation resources a listing of computation resources based on sorting said calculated differences to be said at least a subset of said computation resources for executing said target parallel application process.

20. (Rejected) The computer network of claim 19, wherein said method interfaces to an operating system on said network such that said determination of a listing of computation resources is executed in real time concurrently with said parallel application.

IX. EVIDENCE APPENDIX

(NONE)

X. RELATED PROCEEDINGS APPENDIX

(NONE)